

Optimasi Rute Pengisian Daya Kendaraan Listrik SPKLU Jakarta Pusat Menggunakan Dynamic Programming

Femmy Johan¹, Jennifer Verty², Yohannes³

Jurusan Informatika, Universitas Multi Data Palembang, Jl. Rajawali No.14, 9 Ilir, Kec. Ilir Tim. II, Kota Palembang, Sumatera Selatan^{1,2,3}

femmyjohan_2327250035@mhs.mdp.ac.id¹, jenniferverty_2327250040@mhs.mdp.ac.id²,
yohannesmasterous@mdp.ac.id³

Abstract – The development of electric vehicles in Indonesia has increased the need for Public Electric Vehicle Charging Stations (SPKLU). Electric vehicle users often have difficulty determining the closest SPKLU location and aligning it with the current vehicle position. This study aims to develop an optimization program to determine electric vehicle charging routes at SPKLU in Central Jakarta using the Dynamic Programming algorithm. Research data was obtained from Google Maps, including the SPKLU name, latitude, longitude, wattage, charger type, and address. Distance calculations were performed using the Geodesic Distance method based on the vehicle's coordinates and SPKLU locations. Next, reachable SPKLUs were evaluated using a cost function that combines travel distance and estimated charging time. The Dynamic Programming algorithm was used to determine the minimum-cost solution, and the program then generated a ranking of the best SPKLU recommendations. The study was built in Python and displayed visualizations of locations and routes to SPKLUs on a digital map using Folium. The results showed that the program provided optimal SPKLU recommendations based on a combination of distance and charging time, helping electric vehicle users determine charging locations more effectively and efficiently.

Keywords: *Dynamic Programming, Distance Optimization, Electric Vehicles, Geodesic Distance, SPKLU.*

Abstrak – Perkembangan kendaraan listrik di Indonesia membuat kebutuhan terhadap Stasiun Pengisian Kendaraan Listrik Umum (SPKLU) semakin meningkat. Pengguna kendaraan listrik sering mengalami kesulitan dalam menentukan lokasi SPKLU yang paling dekat dan sesuai dengan posisi kendaraan saat itu. Penelitian ini bertujuan untuk membuat program optimasi penentuan rute pengisian daya kendaraan listrik pada SPKLU di wilayah Jakarta Pusat menggunakan algoritma *Dynamic Programming*. Data penelitian diperoleh dari *Google Maps* yang meliputi nama SPKLU, *latitude*, *longitude*, daya watt, tipe charger dan alamat. Perhitungan jarak dilakukan menggunakan metode *Geodesic Distance* berdasarkan koordinat lokasi kendaraan dan SPKLU. Selanjutnya, SPKLU yang dapat dijangkau akan dievaluasi menggunakan fungsi *cost* yang menggabungkan jarak tempuh dan estimasi waktu pengisian daya. Algoritma *Dynamic Programming* digunakan untuk menentukan nilai *cost* minimum sebagai solusi optimal, kemudian program menghasilkan ranking rekomendasi SPKLU terbaik. Penelitian dibangun menggunakan bahasa pemrograman *Python* dan menampilkan visualisasi lokasi serta rute menuju SPKLU pada peta digital menggunakan *Folium*. Hasil penelitian menunjukkan bahwa program mampu memberikan rekomendasi SPKLU yang optimal berdasarkan kombinasi jarak dan waktu pengisian daya sehingga dapat membantu pengguna kendaraan listrik menentukan lokasi pengisian daya secara lebih efektif dan efisien.

Kata Kunci: *Dynamic Programming, Jarak Geodesic, Kendaraan Listrik, Optimasi Jarak, SPKLU.*

I. PENDAHULUAN

Transportasi merupakan kebutuhan penting bagi kehidupan manusia saat ini. Kemajuan teknologi yang pesat menuntut masyarakat untuk menjadi semakin maju. Perkembangan kendaraan listrik mengalami pertumbuhan yang sangat pesat seiring dengan meningkatnya kesadaran terhadap lingkungan dan upaya global dalam mengurangi emisi karbon. Kendaraan listrik menggunakan energi listrik sebagai sumber utama untuk pengoperasiannya, menggantikan bahan bakar fosil yang selama ini mendominasi sektor transportasi[1]. Penggunaan kendaraan listrik dapat mengurangi dampak negatif terhadap lingkungan dan terdapat potensi penghematan biaya dari penggunaan listrik sebagai bahan bakar daripada bahan bakar fosil[2].

Keberhasilan transisi dari kendaraan konvensional ke kendaraan listrik memerlukan teknik perencanaan rute yang efisien dengan mempertimbangkan karakteristik dari kendaraan listrik seperti jangkauan maksimum kendaraan[2]. Salah satu tantangan utama dalam penggunaan kendaraan listrik adalah keterbatasan jangkauan berkendara (*driving range*) yang sering kali menjadi hambatan bagi pengguna. Keterbatasan ini memunculkan kebutuhan untuk menyediakan stasiun pengisian daya yang cukup untuk memenuhi permintaan yang terus meningkat. Namun, pembangunan stasiun pengisian daya tetap (*fixed charging stations/FCS*) seperti SPKLU membutuhkan investasi yang signifikan dan tidak selalu mampu menangani lonjakan permintaan secara tiba-tiba, sehingga menyebabkan waktu tunggu yang lebih lama bagi pengguna[1].

Maka, untuk masalah ini adalah Optimasi Penentuan Rute Pengisian Daya Kendaraan Listrik sangat yang diperlukan terutama untuk wilayah padat penduduk seperti Jakarta Pusat.

II. TINJAUAN PUSTAKA

1) Kendaraan Listrik

Kendaraan listrik telah menjadi fokus utama dalam industri otomotif di seluruh dunia, terutama karena meningkatnya kesadaran akan pentingnya mengurangi emisi karbon dan ketergantungan pada bahan bakar fosil[3]. Kendaraan listrik merupakan

kendaraan yang menggunakan tenaga listrik sebagai sumber energi untuk beroperasi. Kendaraan listrik menggunakan satu atau lebih motor listrik sebagai penggerak. Motor ini menggunakan energi listrik yang disimpan pada baterai yang dapat diisi ulang ataupun yang dihasilkan secara langsung oleh sel surya atau generator lainnya[1].

Kendaraan listrik sebagai sebuah harapan untuk beralih dari ketergantungan kendaraan berbahan bakar fosil ke bahan bakar alternatif yang lebih ramah lingkungan menggunakan listrik[4].

2) *Dynamic Programming*

Program dinamis atau *dynamic programming* merupakan suatu cara matematis yang digunakan untuk mengambil keputusan dari beberapa serangkaian keputusan yang berkaitan dengan tujuan untuk memudahkan dalam penyelesaian optimasi[5].

Dalam graf, terdapat metode yang dapat digunakan untuk menentukan jarak terpendek. Salah satu metode yang digunakan untuk pencarian jalur terpendek adalah Algoritma *Dynamic Programming*. Algoritma ini digunakan dalam graf berarah dimana setiap titik dihubungkan oleh sisi yang memiliki bobot. Dengan memperhitungkan bobot pada setiap sisi, algoritma ini dapat digunakan untuk menentukan jalur terpendek dari suatu titik ke titik akhir tujuan[6].

Dalam penyelesaian persoalan, program dinamis memiliki dua jenis pendekatan:

1. Program dinamis maju, yaitu bergerak maju mulai dari tahap 1, terus maju ke tahap 2, 3, dan seterusnya sampai tahap n . Runtunan peubah keputusan adalah x_1, x_2, \dots, x_n .
2. Program dinamis mundur, yaitu bergerak mundur mulai tahap n , terus mundur ke tahap $n-1, n-2$, dan seterusnya sampai tahap 1. Runtunan peubah keputusan adalah x_n, x_{n-1}, \dots, x_1 .

Pada praktiknya, terdapat dua jenis program dinamis ditinjau dari data pada persoalannya, yaitu:

1. *Deterministic Dynamic Programming*, yaitu program dinamis untuk menyelesaikan persoalan dengan fungsi ongkos yang terdefinisi dengan jelas dan pasti untuk setiap aksi yang dipilih untuk suatu variabel keputusan.

F. Johan: Optimasi Rute Pengisian Daya ...

2. *Stochastic Dynamic Programming*, yaitu program dinamis untuk menyelesaikan persoalan dengan fungsi ongkos yang memiliki ketidakpastian nilai untuk setiap aksi yang dipilih untuk suatu variabel keputusan. Dalam hal ini, terdapat daftar kemungkinan nilai dan probabilitas untuk nilai tersebut[7].

3) SPKLU

Public Charging Station (PuCS) di Indonesia disebut dengan Stasiun Pengisian Kendaraan Listrik Umum (SPKLU). SPKLU digunakan untuk mengisi kembali daya baterai kendaraan listrik oleh pemilik kendaraan listrik sebagai penunjang mobilitasnya. Berbagai macam konektor khusus untuk pengisian mobil listrik disediakan di *charging station* sesuai dengan standar yang digunakan di dunia. SPKLU dan SPLU (Stasiun Pengisian Listrik Umum) memiliki fungsi yang sama yaitu digunakan untuk pengisian daya. SPLU sendiri terdiri dari 4 (empat) model yaitu model *standing/tower, hang/wall mount, hook/pole mount* dan *stall/pedestal*[1].

Ketersediaan SPKLU belum sebanding dengan kebutuhan pengguna. Ketimpangan ini dipengaruhi oleh kompleksitas penataan ruang Jakarta yang memiliki luas lebih dari 661 km², populasi lebih dari 10 juta jiwa, serta tingkat kemacetan dan aktivitas ekonomi yang sangat tinggi[8].

Fasilitas SPKLU juga mendapatkan saran berupa luasan parkir yang harus mencukupi untuk menampung jumlah kendaraan listrik yang akan melakukan pengisian kendaraan listrik di lokasi, dikarenakan waktu pengisian kendaraan listrik yang memakan waktu, tidak seperti pengisian minyak dari kendaraan berbahan bakar minyak, secara rata-rata, kendaraan listrik melakukan pengisian selama kurang lebih 30 menit untuk stasiun pengisian Fast Charging[9].

4) Optimasi Rute

Optimasi didefinisikan sebagai suatu upaya atau cara untuk mendapatkan penyelesaian (solusi) yang paling baik dari berbagai alternatif penyelesaian yang layak[10]. Model optimasi pada rute transportasi bertujuan untuk meningkatkan efisiensi dengan cara menentukan rute terbaik yang dapat mengurangi biaya, waktu, atau

jarak perjalanan. Model optimasi sering digunakan untuk menyelesaikan berbagai masalah dalam program transportasi[11].

5) Sistem Informasi Geografis

Sistem Informasi Geografis (SIG) merupakan sistem basis data yang dirancang untuk mengelola data yang memiliki referensi keruangan (spasial) serta dilengkapi dengan berbagai fungsi dan operasi untuk pengolahan data tersebut[12]. SIG telah berkembang pesat dalam beberapa dekade terakhir karena berbagai faktor, termasuk peningkatan kemampuan komputasi, kemudahan akses ke data spasial, dan kemajuan perangkat lunak[13]. Data spasial yang dikumpulkan dari berbagai sumber seperti citra satelit peta, data GPS, dan data lainnya, dapat diolah dan dianalisis untuk menghasilkan informasi yang diperlukan dalam pengambilan keputusan tentang penggunaan lahan yang berkelanjutan[14].

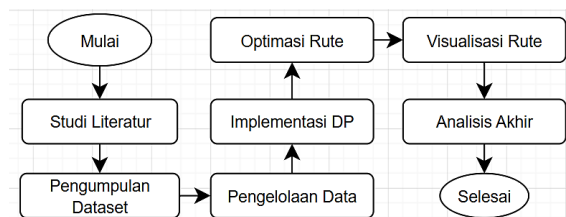
III. ANALISA DAN PERANCANGAN SISTEM

Pada penelitian ini data diperoleh melalui observasi langsung menggunakan *Google Maps* untuk mendapatkan informasi stasiun pengisian kendaraan Listrik umum (SPKLU) di wilayah Jakarta pusat. Data yang dikumpulkana meliputi nama SPKLU, latitude, longitude, daya watt, tipe charger dan alamat.

Variabel yang digunakan pada penelitian ini merupakan Variabel kontinu yang diukur sepanjang kontinum di luar titik desimal[15]. Variabel utama yang digunakan adalah jarak antara lokasi kendaraan listrik dengan lokasi SPKLU, persentase baterai kendaraan, serta koordinat lokasi yang dinyatakan dalam bentuk latitude dan longitude.

A. *Prosedur dan Pengembangan Penelitian*

Prosedur penelitian dalam penelitian ini dapat dilihat pada Gambar 1:



Gambar 1 Flowchart Penelitian

- 1 Menyajikan teori dasar mengenai algoritma *Dynamic Programming* dalam optimasi rute perjalanan.
- 2 Melakukan pengumpulan data lokasi SPKLU di wilayah Jakarta Pusat menggunakan *Google Maps*.
- 3 Menyusun data lokasi SPKLU ke dalam tabel yang berisi nama SPKLU, latitude, longitude, daya watt, tipe charger dan alamat.
- 4 Menghitung jarak antar lokasi menggunakan metode *geodesic distance* berdasarkan koordinat latitude dan longitude.
- 5 Melakukan proses optimasi rute menggunakan algoritma *Dynamic Programming* untuk mencari total jarak minimum.
- 6 Menarik kesimpulan dan memberikan saran pengembangan penelitian.
- 7 Menampilkan hasil optimasi rute serta visualisasi jalur perjalanan menggunakan peta digital berbasis *Folium*.

Pengembangan program dilakukan menggunakan bahasa pemrograman Python dengan pustaka utama yaitu Pandas, Geopy, Request, serta Folium. Pustaka Pandas digunakan untuk mengambil dan mengelola kumpulan data yang berasal dari file CSV. Pustaka Geopy digunakan untuk menghitung jarak antara lokasi kendaraan dan SPKLU berdasarkan koordinat geografis. Pustaka Requests digunakan untuk mengirim permintaan ke layanan API guna memperoleh informasi rute dan lokasi., serta Pustaka Folium digunakan untuk mengintegrasikan data dan visualisasi peta pada antar muka web[16].

IV. IMPLEMENTASI DAN PEMBAHASAN

Implementasi sistem dilakukan menggunakan bahasa pemrograman *Python* dengan beberapa pustaka pendukung, yaitu *Pandas* untuk pengolahan data, *Geopy* untuk menghitung jarak geodesik, dan *Streamlit* sebagai antarmuka pengguna. Sistem menerima masukan berupa koordinat lokasi kendaraan listrik (*latitude* dan *longitude*) serta persentase sisa baterai kendaraan. Selanjutnya, sistem menghitung estimasi energi yang tersisa dan jarak tempuh maksimum kendaraan berdasarkan kapasitas baterai yang tersedia.

Tahap berikutnya adalah melakukan perhitungan jarak antara posisi kendaraan dengan seluruh lokasi SPKLU menggunakan metode *geodesic distance*. SPKLU yang berada di luar jangkauan kendaraan akan dieliminasi, sedangkan SPKLU yang masih dapat dijangkau akan diproses menggunakan algoritma *Dynamic Programming*. Algoritma tersebut digunakan untuk menghitung nilai *cost* berdasarkan kombinasi antara jarak tempuh dan waktu pengisian daya sehingga diperoleh rekomendasi SPKLU dengan nilai biaya minimum.

Hasil akhir sistem menampilkan satu SPKLU terbaik beserta tiga alternatif rekomendasi SPKLU yang telah diurutkan berdasarkan nilai *cost* terkecil. Informasi yang ditampilkan meliputi nama SPKLU, jarak lokasi, tipe charger, daya pengisian, estimasi waktu pengisian, dan alamat SPKLU.

Data Lokasi SPKLU berjumlah 15 telah dicek status operasionalnya diperoleh dari *Google Maps* pada wilayah Jakarta pusat. Data tersebut berisi nama SPKLU, Alamat, latitude, longitude dan lokasi *google maps* dapat dilihat pada Tabel 1.

Dari Tabel 1 dapat dilihat bahwa terdapat beberapa SPKLU yang namanya sama tetapi penempatannya berbeda, data ini didapat berdasarkan sumber *google maps*.

TABEL 1
DATASET PENELITIAN

Nama SPKLU	Lat	Lon	Daya watt	Type charger	Alamat
SPKLU Voltron	-6.22294	106.8032	100	CCS	ARTOTEL Gelora Senayan Jakarta, Jl. Pintu Satu Senayan No.11 1, RT.1/RW.3, Gelora, Kecamatan Tanah Abang, Kota Jakarta Pusat
SPKLU BMW Destination	-6.22114	106.7991	22	Tipe 2	Jl. Asia Afrika, RT.1/RW.3, Gelora, Kecamatan Tanah Abang, Kota Jakarta Pusat
SPKLU Charge+	-6.21296	106.796	60	CCS	Jl. Palmerah Sel. RT.4, RT.4/RW.2, Gelora, Tanah Abang, Central Jakarta City
SPKLU Charge+	-6.20771	106.7947	60	CCS	Jl. Palmerah Barat No.1, RT.1/RW.2, Gelora, Kecamatan Tanah Abang
SPKLU Charge+	-6.19923	106.8229	22	Tipe 2	Jl. M.H. Thamrin No.10, RT.14/RW.20, Kb. Melati, Kecamatan Tanah Abang
SPKLU Voltron	-6.19887	106.8239	100	CCS	The City Tower, Jl. MH Thamrin No.8
SPKLU Terra Charge	-6.19825	106.8408	22	Tipe 2	Menteng Apartments, Jl. Pegangsaan Barat No.Kav.6-12, RT.16/RW.5, Menteng
SPKLU Voltron	-6.19515	106.85	60	CCS	Jl. Salemba Raya No.Kav.31, RT.1/RW.5, Paseban, Kecamatan Senen
SPKLU Voltron	-6.19104	106.8243	22	Tipe 2	Jl. Riau No.5-7, RT.9/RW.5, Gondangdia
SPKLU Voltron	-6.18658	106.823	60	CCS	Jl. M.H. Thamrin No.12, RT.2/RW.1, Kb. Sirih
SPKLU Voltron	-6.17991	106.8222	60	CCS	Jl. Medan Merdeka Barat No.21, RT.2/RW.3, Gambir
SPKLU Voltron	-6.18167	106.835	22	Tipe 2	Jl. Prajurit KKO Usman dan Harun No.44-48, RT.7/RW.1, Gambir
SPKLU Voltron	-6.17246	106.8402	22	Tipe 2	Jl. Senen Raya No.135 137, RW.2, Senen
SPKLU Chargeplus	-6.15082	106.8421	120	CCS	Arena Pekan Raya Jakarta - Kemayoran, Jl. H. Benyamin Sueb No.23, RW.10, Pademangan Timur
SPKLU Voltron	-6.22294	106.8032	100	CCS	ARTOTEL Gelora Senayan Jakarta, Jl. Pintu Satu Senayan No.11 1, RT.1/RW.3, Gelora, Kecamatan Tanah Abang, Kota Jakarta Pusat

A. Proses *Dynamic Programming*

Proses *Dynamic Programming* dilakukan setelah program menghitung jarak antara lokasi kendaraan listrik dan setiap SPKLU yang tersedia. Selanjutnya, program menyeleksi SPKLU yang masih dapat dijangkau berdasarkan estimasi jarak tempuh kendaraan yang diperoleh dari persentase sisa baterai. SPKLU yang memenuhi kriteria keterjangkauan kemudian diproses menggunakan algoritma *Dynamic Programming*. Pada proses ini, setiap SPKLU diberikan nilai *cost* yang dihitung berdasarkan kombinasi jarak menuju SPKLU dan waktu pengisian daya. Nilai *cost* tersebut selanjutnya dibandingkan secara bertahap untuk memperoleh nilai minimum menggunakan pendekatan *Dynamic Programming*.

B. Perhitungan *Cost* SPKLU

Setiap SPKLU yang dapat dijangkau diberikan nilai biaya (*cost*) yang dihitung berdasarkan kombinasi jarak dan waktu pengisian daya:

$$cost_i = jarak_i + (waktu_charger_i \times 10) \quad (1)$$

Keterangan:

- $cost_i$ = nilai biaya SPKLU ke-i
- $jarak_i$ = jarak kendaraan ke SPKLU ke-i (km)
- $waktu_charge_i$ = waktu pengisian daya SPKLU ke-i
- 10 = bobot waktu pengisian

C. Fungsi *Dynamic Programming*

Nilai biaya minimum disimpan menggunakan pendekatan *Dynamic Programming*:

$$dp(i) = \min(dp(i-1), cost_i) \quad (2)$$

Keterangan:

- $dp(i)$ = biaya minimum hingga SPKLU ke-i
- $dp(i-1)$ = biaya minimum sebelumnya
- $cost_i$ = biaya SPKLU saat ini

D. Penyimpanan *Parent*

Untuk menyimpan indeks SPKLU terbaik selama proses optimasi digunakan fungsi:

$$parent(i) = \begin{cases} i, & \text{jika } cost_i < dp(i-1) \\ \text{lainnya} & \text{lainnya} \end{cases} \quad (3)$$

Keterangan:

- $parent(i)$ = indeks SPKLU terbaik sampai tahap ke-i

E. Penelusuran Balik

Pengurutan *cost*:

$$Ranking = sort(cost_i) \quad (4)$$

Keterangan:

- $Ranking$ = daftar SPKLU yang telah diurutkan berdasarkan *cost*
- $cost_i$ = nilai *cost* SPKLU ke-i

Hasil 3 rekomendasi terbaik:

$$hasil_k = SPKLU[Ranking_k], k = 1,2,3 \quad (5)$$

Keterangan:

- $hasil_k$ = rekomendasi SPKLU ke-k
- $Ranking_k$ = indeks SPKLU pada urutan *cost* ke-k

Hasil akhir SPKLU:

$$hasil = SPKLU[best_idx] \quad (6)$$

Keterangan:

- $hasil$ = SPKLU yang direkomendasikan
- $best_idx$ = indeks SPKLU dengan biaya minimum.

Pseudocode *Dynamic Programming*

```

Input: SPKLU terjangkau
for i ← 0 to n-1 do
    cost ← jarak[i] +
(waktu_charge[i] × 10)

    if i = 0 then
        dp[i] ← cost
        parent[i] ← i
    else
        if cost < dp[i-1] then
            dp[i] ← cost
            parent[i] ← i
        else
            dp[i] ← dp[i-1]
            parent[i] ← parent[i-1]
        end if
    end if
end for
  
```

Gambar 2 Pseudocode DP

Dari Gambar 2, dapat dilihat pseudocode *Dynamic Programming* dimulai

dengan menerima input berupa daftar SPKLU yang dapat dijangkau oleh kendaraan listrik. Selanjutnya, algoritma melakukan perulangan dari indeks pertama hingga terakhir untuk menghitung nilai cost setiap SPKLU berdasarkan penjumlahan antara jarak dan waktu pengisian daya yang dikalikan 10. Pada iterasi pertama ($i = 0$), nilai cost langsung disimpan ke dalam array dp dan indeks SPKLU disimpan pada array parent. Pada iterasi berikutnya, nilai cost SPKLU saat ini dibandingkan dengan nilai terbaik sebelumnya yang tersimpan pada dp[i-1]. Jika cost saat ini lebih kecil, maka nilai tersebut disimpan ke dalam dp[i] dan indeksnya disimpan pada parent[i]. Sebaliknya, jika cost lebih besar, maka nilai terbaik sebelumnya dipertahankan dengan menyalin dp[i-1] dan parent[i-1]. Proses ini terus dilakukan hingga seluruh SPKLU selesai diproses.

Pseudocode Penelusuran Balik

```

for i ← 0 to min(2, n-1) do
  idx ← ranking[i].index
  hasil ← SPKLU[idx]

  tampilkan hasil
end for

best_idx ← ranking[0].index
hasil ← SPKLU[best_idx]
index_terpilih ← hasil.index

```

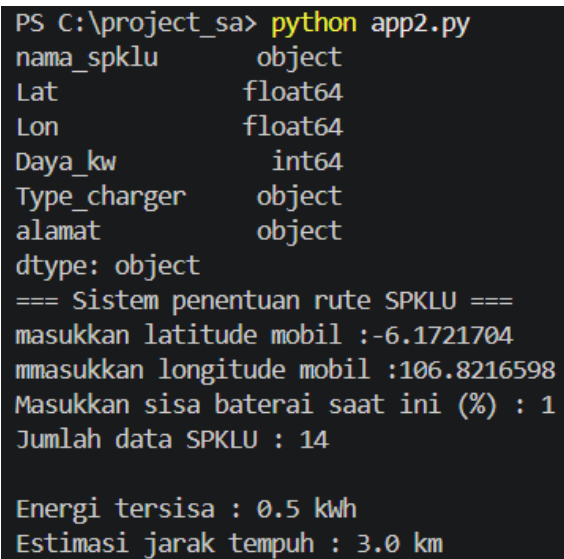
Gambar 4 Pseudocode penelusuran balik

Dari Gambar 3, dapat dilihat pseudocode penelusuran balik digunakan untuk menentukan indeks SPKLU terbaik setelah proses Dynamic Programming selesai. Langkah pertama adalah mencari nilai biaya minimum yang tersimpan pada array dp menggunakan fungsi min(dp). Setelah nilai minimum diperoleh, algoritma melakukan perulangan dari indeks pertama hingga terakhir untuk mencari posisi elemen pada array dp yang memiliki nilai sama dengan min_cost. Ketika nilai tersebut ditemukan, indeksnya disimpan ke dalam variabel best_idx dan proses perulangan dihentikan menggunakan perintah break. Nilai best_idx inilah yang menunjukkan posisi SPKLU dengan biaya total paling rendah sehingga

dapat direkomendasikan sebagai lokasi pengisian daya kendaraan listrik.

F. Inputan Data

Program terlebih dahulu membaca data SPKLU yang terdiri dari nama SPKLU, koordinat latitude dan longitude, daya pengisian (kW), tipe charger, serta alamat lokasi. Selanjutnya, pengguna memasukkan data berupa posisi kendaraan saat ini melalui koordinat latitude dan longitude serta persentase sisa baterai kendaraan.



```

PS C:\project_sa> python app2.py
nama_spklu      object
Lat             float64
Lon            float64
Daya_kw        int64
Type_charger   object
alamat         object
dtype: object
=== Sistem penentuan rute SPKLU ===
masukkan latitude mobil :-6.1721704
mmasukkan longitude mobil :106.8216598
Masukkan sisa baterai saat ini (%) : 1
Jumlah data SPKLU : 14

Energi tersisa : 0.5 kWh
Estimasi jarak tempuh : 3.0 km

```

Gambar 3 Proses Input Data

Pada contoh input di Gambar 4, pengguna memasukkan koordinat kendaraan pada latitude -6.1721704 dan longitude 106.8216598 dengan sisa baterai sebesar 1%. Program kemudian menghitung energi yang masih tersedia pada baterai sebesar 0,5 kWh dan mengestimasi jarak tempuh maksimum yang dapat dicapai kendaraan sejauh 3,0 km. Informasi tersebut digunakan sebagai dasar dalam proses penentuan rute dan pemilihan SPKLU yang dapat dijangkau oleh kendaraan untuk melakukan pengisian daya.

```

=== Proses hitung jarak ===

SPKLU : SPKLU Voltron
Jarak : 5.97 km
Status : Tidak dapat dijangkau

SPKLU : SPKLU BMW Destination
Jarak : 5.96 km
Status : Tidak dapat dijangkau

SPKLU : SPKLU Charge+
Jarak : 5.33 km
Status : Tidak dapat dijangkau

SPKLU : SPKLU Charge+
Jarak : 4.93 km
Status : Tidak dapat dijangkau

SPKLU : SPKLU Charge+
Jarak : 3.0 km
Status : Dapat dijangkau

SPKLU : SPKLU Voltron
Jarak : 2.96 km
Status : Dapat dijangkau

SPKLU : SPKLU Terra Charge
Jarak : 3.58 km
Status : Tidak dapat dijangkau

SPKLU : SPKLU Voltron
Jarak : 4.04 km
Status : Tidak dapat dijangkau

SPKLU : SPKLU Voltron
Jarak : 2.11 km
Status : Dapat dijangkau

SPKLU : SPKLU Voltron
Jarak : 1.6 km
Status : Dapat dijangkau

SPKLU : SPKLU Voltron
Jarak : 0.86 km
Status : Dapat dijangkau

SPKLU : SPKLU Voltron
Jarak : 1.81 km
Status : Dapat dijangkau

SPKLU : SPKLU Voltron
Jarak : 2.05 km
Status : Dapat dijangkau

SPKLU : SPKLU Chargeplus
Jarak : 3.27 km
Status : Tidak dapat dijangkau

```

Gambar 5 Hasil Perhitungan Jarak

TABEL 2
HASIL PERHITUNGAN JARAK

No	Nama SPKLU	Jarak (km)	Status
1	SPKLU Voltron	5.97	Tidak dapat dijangkau
2	SPKLU BMW Destination	5.96	Tidak dapat dijangkau
3	SPKLU Charge+	5.33	Tidak dapat dijangkau
4	SPKLU Charge+	4.93	Tidak dapat dijangkau
5	SPKLU Charge+	3	Dapat dijangkau
6	SPKLU Voltron	2.96	Dapat dijangkau
7	SPKLU Terra Charge	3.58	Tidak dapat dijangkau
8	SPKLU Voltron	4.04	Tidak dapat dijangkau
9	SPKLU Voltron	2.11	Dapat dijangkau
10	SPKLU Voltron	1.6	Dapat dijangkau
11	SPKLU Voltron	0.86	Dapat dijangkau
12	SPKLU Voltron	1.81	Dapat dijangkau
13	SPKLU Voltron	2.05	Dapat dijangkau
14	SPKLU Chargeplus	3.27	Tidak dapat dijangkau

Berdasarkan Gambar 5 dan Tabel 2 mengenai hasil perhitungan jarak, program menghitung jarak antara lokasi kendaraan dan 14 SPKLU menggunakan metode *geodesic distance*. Dengan estimasi jarak tempuh maksimum kendaraan sebesar 3,0 km, diperoleh 7 SPKLU yang dapat dijangkau dan 7 SPKLU yang tidak dapat dijangkau. SPKLU terdekat memiliki jarak 0,86 km, sedangkan yang terjauh berjarak 5,97 km. Selanjutnya, hanya SPKLU yang dapat dijangkau yang dipilih untuk diproses pada tahap optimasi menggunakan algoritma *Dynamic Programming*.

TABEL 3
HASIL PROSES DP

State	Nama SPKLU	Cost	DP	Parent	Rumus DP
1	SPKLU Charge+	25.5	25.5	0	$f[0] = 25.50$
2	SPKLU Voltron	7.91	7.91	1	$f[1] = \min(f[0], 7.91)$
3	SPKLU Voltron	24.61	7.91	1	$f[2] = \min(f[1], 24.61)$
4	SPKLU Voltron	9.85	7.91	1	$f[3] = \min(f[2], 9.85)$
5	SPKLU Voltron	9.11	7.91	1	$f[4] = \min(f[3], 9.11)$
6	SPKLU Voltron	24.31	7.91	1	$f[5] = \min(f[4], 24.31)$
7	SPKLU Voltron	24.55	7.91	1	$f[6] = \min(f[5], 24.55)$

```

=== Proses dynamic programming ===

State ke-1
SPKLU : SPKLU Charge+
Cost : 25.5
DP : 25.5
Parent : 0
f[0] = 25.5

State ke-2
SPKLU : SPKLU Voltron
Cost : 7.91
DP : 7.91
Parent : 1
f[1] = min(f[0], 7.91)

State ke-3
SPKLU : SPKLU Voltron
Cost : 24.61
DP : 7.91
Parent : 1
f[2] = min(f[1], 24.61)

State ke-4
SPKLU : SPKLU Voltron
Cost : 9.85
DP : 7.91
Parent : 1
f[3] = min(f[2], 9.85)

State ke-5
SPKLU : SPKLU Voltron
Cost : 9.11
DP : 7.91
Parent : 1
f[4] = min(f[3], 9.11)

State ke-6
SPKLU : SPKLU Voltron
Cost : 24.31
DP : 7.91
Parent : 1
f[5] = min(f[4], 24.31)

State ke-7
SPKLU : SPKLU Voltron
Cost : 24.55
DP : 7.91
Parent : 1
f[6] = min(f[5], 24.55)
    
```

Gambar 6 Hasil Proses DP

Berdasarkan Gambar 6 dan Tabel 3 hasil proses DP, setiap SPKLU yang dapat dijangkau dihitung nilai *cost*-nya berdasarkan kombinasi jarak tempuh dan estimasi waktu pengisian daya. Pada state pertama, nilai *cost* sebesar 25,50 menjadi nilai awal DP. Selanjutnya, algoritma membandingkan *cost* pada setiap *state* dengan nilai DP sebelumnya menggunakan fungsi minimum.

Pada state ke-2, SPKLU Voltron memiliki *cost* sebesar 7,91 yang lebih kecil dari nilai sebelumnya sehingga menjadi nilai DP optimal dan disimpan sebagai solusi terbaik. Hingga state ke-7, tidak ada *cost* yang lebih kecil dari 7,91 sehingga nilai DP tetap sama. Hasil ini menunjukkan bahwa SPKLU Voltron pada state ke-2 merupakan alternatif terbaik dengan nilai *cost* minimum sebesar 7,91.

```

=== Penelusuran Balik ===

Hasil ke-1
SPKLU : SPKLU Voltron
Alamat : The City Tower, Jl. MH Thamrin No.8
Cost : 7.91
Jarak : 2.96 km
Daya : 100 kW

Hasil ke-2
SPKLU : SPKLU Voltron
Alamat : Jl. Medan Merdeka Barat No.21, RT.2/RW.3, Gambir
Cost : 9.11
Jarak : 0.86 km
Daya : 60 kW

Hasil ke-3
SPKLU : SPKLU Voltron
Alamat : Jl. M.H. Thamrin No.12, RT.2/RW.1, Kb. Sirih
Cost : 9.85
Jarak : 1.6 km
Daya : 60 kW
    
```

Gambar 7 Hasil Penelusuran Balik

TABEL 4
HASIL PENELUSURAN BALIK

Hasil	Nama SPKLU	Alamat	Cost	Jarak (km)	Daya (kW)
1	SPKLU Voltron	The City Tower, Jl. MH Thamrin No.8	7.91	2.96	100
2	SPKLU Voltron	Jl. Medan Merdeka Barat No.21, RT.2/RW.3, Gambir	9.11	0.86	60
3	SPKLU Voltron	Jl. M.H. Thamrin No.12, RT.2/RW.1, Kb. Sirih	9.85	1.6	60

TABEL 5
HASIL RANKING SPKLU

Ranking	Nama SPKLU	Alamat	Cost
1	SPKLU Voltron	The City Tower, Jl. MH Thamrin No.8	7.91
2	SPKLU Voltron	Jl. Medan Merdeka Barat No.21, RT.2/RW.3, Gambir	9.11
3	SPKLU Voltron	Jl. M.H. Thamrin No.12, RT.2/RW.1, Kb. Sirih	9.85
4	SPKLU Voltron	Jl. Prajurit KKO Usman dan Harun No.44-48, RT.7/RW.1, Gambir	24.31
5	SPKLU Voltron	Jl. Senen Raya No.135-137, RW.2, Senen	24.55
6	SPKLU Voltron	Jl. Riau No.5-7, RT.9/RW.5, Gondangdia	24.61
7	SPKLU Charge+	Jl. M.H. Thamrin No.10, RT.14/RW.20, Kb. Melati, Kecamatan Tanah Abang	25.50

Berdasarkan hasil penelusuran balik pada Gambar 7 dan Tabel 4, diperoleh tiga rekomendasi SPKLU terbaik yang telah diurutkan berdasarkan nilai cost terkecil. SPKLU Voltron yang berlokasi di The City Tower, Jl. MH Thamrin No.8 menempati peringkat pertama dengan nilai cost sebesar 7,91 sehingga menjadi solusi optimal yang direkomendasikan. Peringkat kedua dan ketiga juga ditempati oleh SPKLU Voltron dengan nilai cost masing-masing 9,11 dan 9,85. Hasil ini menunjukkan bahwa ketiga SPKLU tersebut merupakan alternatif pengisian daya yang paling optimal berdasarkan kombinasi jarak tempuh kendaraan dan estimasi waktu pengisian daya yang dihitung oleh algoritma *Dynamic Programming*.

```

=== Ranking SPKLU Terbaik ===

Ranking 1
SPKLU : SPKLU Voltron
Alamat : The City Tower, Jl. MH Thamrin No.8
Cost : 7.91

Ranking 2
SPKLU : SPKLU Voltron
Alamat : Jl. Medan Merdeka Barat No.21, RT.2/RW.3, Gambir
Cost : 9.11

Ranking 3
SPKLU : SPKLU Voltron
Alamat : Jl. M.H. Thamrin No.12, RT.2/RW.1, Kb. Sirih
Cost : 9.85

Ranking 4
SPKLU : SPKLU Voltron
Alamat : Jl. Prajurit KKO Usman dan Harun No.44-48, RT.7/RW.1, Gambir
Cost : 24.31

Ranking 5
SPKLU : SPKLU Voltron
Alamat : Jl. Senen Raya No.135 137, RW.2, Senen
Cost : 24.55

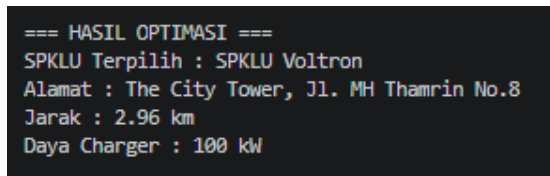
Ranking 6
SPKLU : SPKLU Voltron
Alamat : Jl. Riau No.5-7, RT.9/RW.5, Gondangdia
Cost : 24.61

Ranking 7
SPKLU : SPKLU Charge+
Alamat : Jl. M.H. Thamrin No.10, RT.14/RW.20, Kb. Melati, Kecamatan Tanah Abang
Cost : 25.5

```

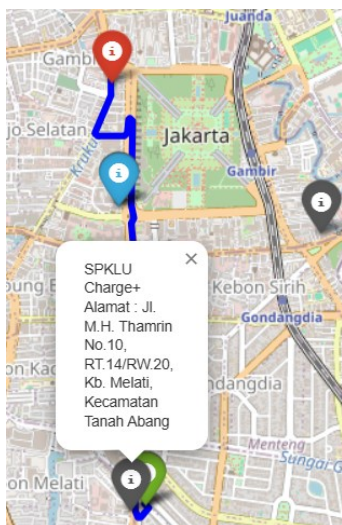
Gambar 8 Hasil Ranking SPKLU

Berdasarkan hasil perankingan pada Gambar 8 dan Tabel 5, SPKLU diurutkan berdasarkan nilai cost dari yang terkecil hingga terbesar. SPKLU Voltron yang berlokasi di The City Tower, Jl. MH Thamrin No.8 menempati peringkat pertama dengan nilai cost sebesar 7,91, sehingga menjadi rekomendasi terbaik untuk dipilih. Peringkat kedua dan ketiga juga ditempati oleh SPKLU Voltron dengan nilai cost masing-masing 9,11 dan 9,85. Sementara itu, SPKLU Charge+ berada pada peringkat terakhir dengan nilai cost sebesar 25,50. Hasil perankingan ini menunjukkan bahwa SPKLU dengan nilai *cost* yang lebih rendah lebih diprioritaskan karena memberikan kombinasi jarak tempuh dan waktu pengisian daya yang lebih optimal bagi pengguna kendaraan listrik.



Gambar 9 Hasil Optimasi Program

Dari Gambar 9, hasil optimasi berupa SPKLU yang direkomendasikan oleh program. Sebelum menentukan SPKLU terpilih, program terlebih dahulu melakukan proses seleksi terhadap seluruh SPKLU yang berada dalam jangkauan kendaraan berdasarkan estimasi jarak tempuh dari sisa baterai.



Gambar 10 Visualisasi Maps

Dari Gambar 10, dapat dilihat program memberikan visualisasi maps output informasi ranking dan letak SPKLU beserta jarak dengan kendaraan, sehingga *user* atau pengguna dapat langsung menuju ke SPKLU terdekat.

V. PENGUJIAN

Pengujian sistem dilakukan untuk mengetahui kemampuan program dalam memberikan rekomendasi SPKLU yang optimal pada berbagai kondisi persentase baterai dan lokasi kendaraan. Pengujian dilakukan dengan mengubah nilai input persentase baterai sehingga menghasilkan variasi jangkauan kendaraan yang berbeda.

TABEL 6
PENGUJIAN

No	Koordinat Kendaraan	Persentase Baterai	Estimasi Jarak tempuh	Hasil Pengujian
1	(-6.1760454 , 106.8237019)	2%	6 km	Seluruh SPKLU dapat dijangkau
2	(-6.2070288, 106.8005641)	1%	3km	6 SPKLU dapat dijangkau
3	(-6.209383, 106.7925779)	0,5%	1,5 km	3 SPKLU dapat dijangkau

Berdasarkan Tabel 6, dapat dilihat bahwa semakin besar persentase baterai kendaraan, maka semakin banyak SPKLU yang dapat dijangkau. Kondisi ini menunjukkan bahwa sistem mampu melakukan proses penyaringan lokasi SPKLU secara dinamis sesuai dengan kondisi kendaraan listrik yang dimasukkan oleh pengguna.

Selain itu, perubahan koordinat kendaraan juga memengaruhi hasil rekomendasi SPKLU. Sistem akan selalu memilih SPKLU dengan nilai *cost* terkecil berdasarkan kombinasi antara jarak tempuh dan waktu pengisian daya. Dengan demikian, algoritma *Dynamic Programming* mampu memberikan alternatif SPKLU yang lebih

efektif dan efisien sesuai kondisi kendaraan secara *real-time*.

VI. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan, program rekomendasi SPKLU di Jakarta Pusat menggunakan algoritma *Dynamic Programming* berhasil dibangun dan mampu menentukan SPKLU terbaik yang dapat dijangkau oleh kendaraan listrik berdasarkan sisa baterai yang tersedia. Program menghitung jarak menggunakan metode *geodesic distance*, kemudian melakukan optimasi berdasarkan nilai *cost* yang merupakan kombinasi jarak dan estimasi waktu pengisian daya untuk memperoleh SPKLU dengan biaya minimum. Pengembangan program selanjutnya dapat dilakukan dengan menambahkan faktor lain seperti kondisi lalu lintas, biaya pengisian daya, dan tingkat antrian pada SPKLU agar hasil rekomendasi menjadi lebih akurat.

Selain itu, penelitian berikutnya dapat membandingkan *Dynamic Programming* dengan algoritma lain seperti Dijkstra atau A* untuk memperoleh hasil optimasi yang lebih baik.

REFERENSI

- [1] D. A. H. Indrawati, "Model Optimasi Stasiun Pengisian Daya Bergerak Untuk Kendaraan Listrik," *Perpust. Digit. ITB*, p. 107, 2024, [Online]. Available: https://digilib.itb.ac.id/gdl/view_data/model-optimasi-stasiun-pengisian-daya-bergerak-untuk-kendaraan-listrik/Model-Optimasi-Stasiun-Pengisian-Daya-Bergerak-Untuk-Kendaraan-Listrik?rows=1&per_page=2
- [2] Z. I. Haryanto and A. A. A. Putri, "Analisis Perbandingan Strategi Pengisian Ulang pada Masalah Electric Vehicle Routing Problem with Time Windows (EVRPTW): Studi Kasus Layanan Ekspedisi," *J. Ilm. Mhs. Tek. Ind.*, vol. 7, no. 1, pp. 44–58, 2024, doi: 10.30737/JURMATIS.V7I1.6291.
- [3] R. Merdiansah and A. A. Ridha, "Analisis Sentimen Pengguna X Indonesia Terkait Kendaraan Listrik Menggunakan IndoBERT," *J. Ilmu Komput. dan Sist. Inf.*, vol. 7, no. 1, pp. 221–228, 2024, doi: 10.55338/jikomsi.v7i1.2895.
- [4] G. Zola *et al.*, "Inovasi kendaraan listrik sebagai upaya meningkatkan kelestarian lingkungan dan mendorong pertumbuhan ekonomi hijau di Indonesia," *e-Jurnal Ekon. Sumberd. dan Lingkung.*, vol. 11, no. 3, pp. 159–170, 2023, doi: 10.22437/jesl.v12i3.30229.
- [5] K. G. Sudarman, "Optimalisasi Biaya Distribusi dan Kapasitas Pengiriman Produk Kopi Menggunakan Metode Dynamic Programming Dengan Software Win-QSB Pada PT MLN," *Pros. Konsorsium Semin. Nas. Waluyo Jatmiko 2025*, vol. 18, no. 1, pp. 145–154, 2025, doi: 10.33005/wj.v18i1.156.
- [6] S. W. Ginting, E. A. Dharmawan, and R. P. J. V. Nikijuluw, "Pencarian Rute Optimum Antar Objek Wisata Di Jazirah Leihitu Barat Menggunakan Algoritma Dynamic Programming," *J. IT Media Inf. STMIK Handayani Makassar*, vol. 15, no. 3, pp. 173–181, 2024, doi: 10.37639/jti.v15i3.391.
- [7] A. Thariq, "Perbandingan Penerapan Algoritma Dynamic Programming dengan Algoritma Greddy dalam Menentukan Optimasi Posisi Pasar di Suatu Wilayah," *J. Simetrik (Sipil, Mesin, List.*, vol. 13, no. 1, pp. 690–696, 2023, doi: 10.31959/js.v13i1.1316.
- [8] A. S. Lesmana and I. K. Gunarta, "Analisis Pemilihan Lokasi Stasiun Pengisian Kendaraan Listrik Umum (SPKLU) di DKI Jakarta Menggunakan Geographic Information System dan Analytical Hierarchy Process," *J. Impresi Indones.*, vol. 4, no. 12, pp. 5474–5481, 2025, doi: 10.58344/jii.v4i12.7258.
- [9] A. J. Atthariq, D. Puspitasari, R. Panjaitan, and A. Anomsari, "Pengaruh Persepsi Harga, Peran Insentif Pemerintah, Dan Ketersediaan Stasiun Pengisian Kendaraan Listrik Umum (SPKLU) Terhadap Keputusan Pembelian Kendaraan Listrik Kota Semarang," *Din. J. Manaj. Sos. Ekon.*, vol. 4, no. 2, pp. 9–23, 2024, doi: 10.51903/b826wz61.
- [10] W. Y. Oeitama and W. Y. Oeitama, "Optimasi Rute Pendistribusian Barang Menggunakan Kombinasi Algoritma Branch and Bound dan Cheapest Insertion Heuristic," *Sq. J. Math. Math. Educ.*, vol. 6, no. 2, pp. 89–104, 2024, doi: 10.21580/square.2024.6.2.22992.
- [11] H. W. N. S *et al.*, "Model Optimasi Rute Transportasi Berbasis Pemrograman Linear," *J. Sist. Inf. Triguna Dharma*

- (*JURSI TGD*), vol. 4, no. 1, pp. 75–81, 2025, doi: 10.53513/jursi.v4i1.10586.
- [12] Y. Rehan, “Sistem Informasi Geografis Persebaran Lembaga Kursus Bahasa Inggris Berbasis Web (Studi Kasus Kampung Inggris Kecamatan Pare Kabupaten Kediri),” *J. Inform. dan Multimed.*, vol. 14, no. 1, pp. 34–45, 2022, doi: 10.33795/jim.v14i1.360.
- [13] F. T. Prasetyo and A. W. Utami, “Rancang Bangun Sistem Informasi Geografis Untuk Optimasi Rute Terpendek Distribusi Produk Dengan Algoritma A-Star (Studi Kasus UD Ali Berkah),” *J. Emerg. Inf. Syst. Bus. Intell.*, vol. 05, no. 03, pp. 181–190, 2024, doi: 10.26740/jeisbi.v5i3.62919.
- [14] M. A. Lasaiba and D. S. Alnursa, “Pengolahan Data Spasial dalam Perencanaan Penggunaan Lahan yang Berkelanjutan,” *GEOFORUM J. Geogr. dan Pendidik. Geogr.*, vol. 2, no. 1, pp. 1–12, 2023, doi: 10.30598/geoforumvol2iss1pp1-12.
- [15] M. Tarigan and D. F. Silaban, “Reviu Statistika: Data Dan Skala Pengukuran Statistika Dalam Keperawatan,” *JINTAN J. Ilmu Keperawatan*, vol. 3, no. 2, pp. 118–126, 2023, doi: 10.51771/jintan.v3i02.658.
- [16] R. Priambodo, H. Hermawan, L. Alfat, and T. M. Kadarina, “Pengembangan Peta Interaktif Layanan Puskesmas Nasional melalui Integrasi Open Data dengan Streamlit dan Folium,” *J. Ilm. FIFO*, vol. 17, no. 2, pp. 144–152, 2025, doi: 10.22441/fifo.2025.v17i2.004.